TEIDRVR.ASM
ROM AT F400

```
                        ;
                        ;       Driver for TEI video board
                        ;
                        ;
                        ;       tei video board i/o control ports
                        ;
0090 =          vhlncnt equ     90h
0091 =          vhsynwt equ     vhlncnt+1
0092 =          vscnrow equ     vhlncnt+2
0093 =          vskwbit equ     vhlncnt+3
0094 =          vscnlin equ     vhlncnt+4
0095 =          vvrtdat equ     vhlncnt+5
0096 =          vlstdrw equ     vhlncnt+6
0099 =          vrchr   equ     vhlncnt+9
0098 =          vrlin   equ     vhlncnt+8
009A =          vrset   equ     vhlncnt+0ah
009B =          vscrol  equ     vhlncnt+0bh
009C =          vlchr   equ     vhlncnt+0ch
009D =          vllin   equ     vhlncnt+0dh
009E =          vstrtim equ     vhlncnt+0eh
                        ;
                        ;       initialization data values for tei video board
                        ;
0064 =          dhlncnt equ     100
0043 =          dhsynwt equ     01000011b
0045 =          dscnrow equ     01000101b
0057 =          dskwbit equ     01010111b
0003 =          dscnlin equ     3
001E =          dvrtdat equ     30
0017 =          dlstdrw equ     23
                        ;
                        ;       video memory origin
                        ;       and screen size parameters
                        ;
E800 =          vbase   equ     0e800h
0018 =          vmxlin  equ     24
0050 =          vmxchr  equ     80
                        ;
                        ;
                        ;
F400                    org     0f400h
                        ;
                        ;       Entry vectors
                        ;
F400 C306F4             jmp     vreset
F403 C338F4             jmp     vout
                        ;
                        ;       routine to reset video board
                        ;
F406 =          vreset  equ     $
F406 3E64               mvi     a,dhlncnt
F408 D390               out     vhlncnt
```

```
F40A 3E43               mvi     a,dhsynwt
F40C D391               out     vhsynwt
F40E 3E45               mvi     a,dscnrow
F410 D392               out     vscnrow
F412 3E57               mvi     a,dskwbit
F414 D393               out     vskwbit
F416 3E03               mvi     a,dscnlin
F418 D394               out     vscnlin
F41A 3E1E               mvi     a,dvrtdat
F41C D395               out     vvrtdat
F41E 3E17               mvi     a,dlstdrw
F420 D396               out     vlstdrw
F422 D39E               out     vstrtim
F424 AF                 xra     a
F425 D39D               out     vllin
F427 3C                 inr     a
F428 D39C               out     vlchr
F42A AF                 xra     a
F42B 3282EF             sta     vspeed
F42E 3283EF             sta     vescflg
F431 3281EF             sta     vspeed-1
F434 3284EF             sta     vphyl0
F437 C9                 ret
                ;
                ;       tei video board output dispatcher
                ;
F438 =          vout    equ     $
F438 E5                 push    h               ;save regs
F439 D5                 push    d
F43A C5                 push    b
F43B 3A83EF             lda     vescflg         ;in escape?
F43E B7                 ora     a
F43F C25AF5             jnz     vescs           ;yes
F442 78                 mov     a,b             ;no, get char
F443 E67F               ani     7fh             ;strip parity
F445 211AF6             lxi     h,vtbl          ;special char table
F448 CD59F4             call    vtsrch
                vback1:
F44B 2A81EF             lhld    vspeed-1        ;delay loop
F44E 2C                 inr     l
F44F AF                 xra     a
                vtimer:
F450 2B                 dcx     h
F451 BC                 cmp     h
F452 C250F4             jnz     vtimer
F455 =          vgobk   equ     $
F455 C1                 pop     b               ;restore regs
F456 D1                 pop     d
F457 E1                 pop     h
F458 C9                 ret
                ;
                ;       special character lookup/dispatcher
                ;
F459 =          vtsrch  equ     $
```

```
F459 7E                 mov     a,m
F45A B7                 ora     a
F45B CA6FF4             jz      voutr           ;ordinary character
F45E B8                 cmp     b
F45F 23                 inx     h
F460 CA68F4             jz      vtsrch2         ;hit
F463 23                 inx     h
F464 23                 inx     h
F465 C359F4             jmp     vtsrch
                vtsrch2:
F468 7E                 mov     a,m
F469 23                 inx     h
F46A 66                 mov     h,m
F46B 6F                 mov     l,a
F46C E5                 push    h               ;table ptr now return addr
F46D 7D                 mov     a,l
F46E C9                 ret
                ;
                ;       output char in b and bump cursor
                ;
F46F =          voutr   equ     $
F46F CD84F4             call    vchout
F472 DB99               in      vrchr           ;char posn
F474 FE50               cpi     vmxchr          ;at end?
F476 CA7DF4             jz      vcrlf           ;yes
F479 3C                 inr     a               ;nope
F47A D39C               out     vlchr
F47C C9                 ret
F47D =          vcrlf   equ     $
F47D CDF2F4             call    vcr
F480 CDF7F4             call    vlf
F483 C9                 ret
                ;
                ;       output 1 char at current cursor posn
                ;
F484 =          vchout  equ     $
F484 CD10F6             call    vcadd           ;memory address
F487 70                 mov     m,b             ;stuff it
F488 C9                 ret
                ;
                ;       clear screen and home cursor
                ;
F489 =          vclear  equ     $
F489 2100E8             lxi     h,vbase
F48C 018007             lxi     b,vmxlin*vmxchr
                vclear1:
F48F 3620               mvi     m,' '
F491 23                 inx     h
F492 0B                 dcx     b
F493 78                 mov     a,b
F494 B1                 ora     c
F495 C28FF4             jnz     vclear1
                ;
                ;       home cursor
```

```
                        ;
F498 =          vhome   equ     $
F498 AF                 xra     a
F499 3280EF             sta     vclrscn         ;zero curent line #
F49C 3A84EF             lda     vphyl0          ;board's line 0
F49F D39D               out     vllin
F4A1 3E01               mvi     a,1             ;character 1
F4A3 D39C               out     vlchr
F4A5 C9                 ret
                        ;
                        ;       move cursor up one line with wrap
                        ;
F4A6 =          vup     equ     $
F4A6 DB98               in      vrlin           ;get current line
F4A8 B7                 ora     a               ;at top?
F4A9 C2AEF4             jnz     vup1            ;nope
F4AC 3E18               mvi     a,vmxlin        ;yes - to bottom
                vup1:
F4AE 3D                 dcr     a               ;up 1 line
F4AF D39D               out     vllin           ;to board
F4B1 3A80EF             lda     vclrscn         ;get internal line #
F4B4 B7                 ora     a               ;0?
F4B5 C2BAF4             jnz     vup2            ;nope
F4B8 3E18               mvi     a,vmxlin        ;yes, set to max
                vup2:
F4BA 3D                 dcr     a               ;up 1 line
F4BB 3280EF             sta     vclrscn         ;set internal line #
F4BE C9                 ret
                        ;
                        ;       move cursor down 1 line with wrap
                        ;
F4BF =          vdown   equ     $
F4BF DB98               in      vrlin           ;get current line
F4C1 3C                 inr     a               ;down 1
F4C2 FE18               cpi     vmxlin          ;at max?
F4C4 C2C8F4             jnz     vdown1          ;nope
F4C7 AF                 xra     a               ;to top
                vdown1:
F4C8 D39D               out     vllin           ;tell board
F4CA 3A80EF             lda     vclrscn         ;get internal line #
F4CD 3C                 inr     a               ;down 1
F4CE FE18               cpi     vmxlin          ;at max?
F4D0 C2D4F4             jnz     vdown2          ;nope
F4D3 AF                 xra     a               ;set to top
                vdown2:
F4D4 3280EF             sta     vclrscn         ;set new internal line #
F4D7 C9                 ret
                        ;
                        ;       move cursor left with wrap
                        ;
F4D8 =          vleft   equ     $
F4D8 DB99               in      vrchr           ;current char
F4DA 3D                 dcr     a               ;left 1
F4DB FE00               cpi     0               ;off end?
```

```
F4DD C2E2F4              jnz     vleft1          ;nope
F4E0 3E50                mvi     a,vmxchr        ;to right side
                 vleft1:
F4E2 D39C                out     vlchr
F4E4 C9                  ret
                 ;
                 ;       move cursor right with wrap
                 ;
F4E5 =           vright  equ     $
F4E5 DB99                in      vrchr           ;current char
F4E7 3C                  inr     a               ;right 1
F4E8 FE51                cpi     vmxchr+1        ;off end?
F4EA C2EFF4              jnz     vright1         ;nope
F4ED 3E01                mvi     a,1             ;to left side
                 vright1:
F4EF D39C                out     vlchr
F4F1 C9                  ret
                 ;
                 ;       carriage return
                 ;
F4F2 =           vcr     equ     $
F4F2 3E01                mvi     a,1
F4F4 D39C                out     vlchr           ;column 1
F4F6 C9                  ret
                 ;
                 ;       line feed
                 ;
F4F7 =           vlf     equ     $
F4F7 AF                  xra     a               ;clear clear flag
F4F8 3285EF              sta     vlfclr
F4FB 3A80EF              lda     vclrscn         ;current internal line #
F4FE FE17                cpi     vmxlin-1        ;on last line?
F500 C21BF5              jnz     vlf1            ;no, no scroll
F503 D39B                out     vscrol          ;oops, scroll time...
F505 3EFF                mvi     a,0ffh          ;set clear flag
F507 3285EF              sta     vlfclr
F50A 3A84EF              lda     vphyl0          ;get current board line 0
F50D 3C                  inr     a               ;bump 1
F50E FE18                cpi     vmxlin          ;past end?
F510 DA15F5              jc      vlfa            ;nope
F513 D618                sui     vmxlin          ;yes, put it back
                 vlfa:
F515 3284EF              sta     vphyl0          ;new physical line 0
F518 C31FF5              jmp     vlf2
                 vlf1:
F51B 3C                  inr     a               ;bump current internal line #
F51C 3280EF              sta     vclrscn
                 vlf2:
F51F DB98                in      vrlin           ;read board line #
F521 3C                  inr     a               ;down 1
F522 FE18                cpi     vmxlin          ;off end?
F524 C228F5              jnz     vlf3            ;nope
F527 AF                  xra     a               ;yes, back to top
                 vlf3:
```

```
F528 D39D               out     vllin           ;set board line #
F52A 3A85EF             lda     vlfclr          ;need to clear?
F52D B7                 ora     a
F52E C4C8F5             cnz     vclrlin         ;clear line
F531 C9                 ret
                ;
                ;       move left 1 char and erase
                ;
F532 =          vbacks  equ     $
F532 CDD8F4             call    vleft
F535 CD10F6             call    vcadd           ;get memory address
F538 3620               mvi     m,' '           ;erase char
F53A C9                 ret
                ;
                ;       erase to end of line
                ;
F53B =          veeol   equ     $
F53B DB99               in      vrchr           ;current column #
F53D F5                 push    psw             ;save it
                veeol1:
F53E FE51               cpi     vmxchr+1        ;at end?
F540 CA50F5             jz      veeol2          ;yes, done
F543 F5                 push    psw             ;save column #
F544 D39C               out     vlchr           ;change board column
F546 0620               mvi     b,' '
F548 CD84F4             call    vchout
F54B F1                 pop     psw             ;restore column #
F54C 3C                 inr     a               ;next column
F54D C33EF5             jmp     veeol1
                veeol2:
F550 F1                 pop     psw             ;restore original column
F551 D39C               out     vlchr           ;tell board
F553 C9                 ret
                ;
                ;       escape sequence, first char
                ;
F554 =          vesc    equ     $
F554 3EFF               mvi     a,0ffh          ;set second char esc flag
F556 3283EF             sta     vescflg
F559 C9                 ret
                ;
                ;       escape sequence, 2nd and 3rd chars
                ;
F55A =          vescs   equ     $
F55A CD60F5             call    vescsp          ;do esape processing
F55D C355F4             jmp     vgobk           ;and return
                vescsp:
F560 3A83EF             lda     vescflg         ;2nd or 3rd?
F563 FEFF               cpi     0ffh
F565 CA84F5             jz      vesc2           ;2nd
F568 2183EF             lxi     h,vescflg       ;3rd, reset flag
F56B 7E                 mov     a,m             ;get 2nd char
F56C 3600               mvi     m,0
F56E FE02               cpi     2               ;was it 2
```

```
F570 DA9DF5             jc      vsetx           ;nope, 1
F573 CAA4F5             jz      vsety           ;yes, 2
F576 FE08               cpi     8               ;was it 8
F578 CAC3F5             jz      vsetspd         ;yes
F57B FE09               cpi     9               ;6?
F57D DA6FF4             jc      voutr           ;yes
F580 C0                 rnz                     ;not 9 either, ignore
F581 C39DF5             jmp     vsetx           ;9 is like 1
                    vesc2:
F584 78                 mov     a,b             ;code in a
F585 FE03               cpi     3               ;is it 3?
F587 CAB9F5             jz      vcuret          ;yes
F58A FE04               cpi     4               ;4?
F58C C299F5             jnz     veret2          ;no, just save code
                    ;
                    ;       return char memory address
                    ;
F58F CD10F6             call    vcadd
                    ;
                    ;       escape sequence return processing
                    ;
F592 =              veret   equ     $
F592 44                 mov     b,h             ;put hl in bc to save on stack
F593 4D                 mov     c,l
F594 =              veret1  equ     $               ;replace b on stack
F594 E1                 pop     h               ;return address
F595 D1                 pop     d               ;old bc
F596 C5                 push    b
F597 E5                 push    h               ;return address
F598 AF                 xra     a               ;end of escape
                    veret2:
F599 3283EF             sta     vescflg
F59C C9                 ret
                    ;
                    ;       set x position
                    ;
F59D =              vsetx   equ     $
F59D 78                 mov     a,b
F59E E67F               ani     7fh
F5A0 3C                 inr     a               ;relative 1
F5A1 D39C               out     vlchr
F5A3 C9                 ret
                    ;
                    ;       set y position
                    ;
F5A4 =              vsety   equ     $
F5A4 78                 mov     a,b
F5A5 E67F               ani     7fh
F5A7 3280EF             sta     vclrscn         ;set current line #
F5AA 4F                 mov     c,a             ;save desired line #
F5AB 3A84EF             lda     vphyl0          ;find physical line 0
F5AE 81                 add     c               ;bump to desired line
F5AF FE18               cpi     vmxlin          ;off the end?
F5B1 DAB6F5             jc      vsetyl          ;nope
```

```
F5B4 D618               sui     vmxlin          ;mod vmxlin
                vsety1:
F5B6 D39D               out     vllin
F5B8 C9                 ret
                ;
                ;       return row column
                ;
F5B9 =          vcuret  equ     $
F5B9 DB98               in      vrlin
F5BB 4F                 mov     c,a
F5BC DB99               in      vrchr
F5BE 3D                 dcr     a               ;relative 0
F5BF 47                 mov     b,a
F5C0 C394F5             jmp     veret1
                ;
                ;       set speed
                ;
F5C3 =          vsetspd equ     $
F5C3 78                 mov     a,b
F5C4 3282EF             sta     vspeed
F5C7 C9                 ret
                ;
                ;       routine to clear current line
                ;
F5C8 =          vclrlin equ     $
F5C8 CDDFF5             call    vladd           ;get line address
F5CB 113000             lxi     d,30h           ;end of 1 group to start of next

F5CE 0E05               mvi     c,vmxchr/16     ;number of groups
                vclin1:
F5D0 0610               mvi     b,16            ;group size
                vclin2:
F5D2 3620               mvi     m,' '           ;clear char
F5D4 23                 inx     h               ;bump ptr
F5D5 05                 dcr     b               ;decrement char within group
F5D6 C2D2F5             jnz     vclin2          ;still in group
F5D9 19                 dad     d               ;bump to next group
F5DA 0D                 dcr     c               ;decrement group ctr
F5DB C2D0F5             jnz     vclin1          ;not done yet
F5DE C9                 ret
                ;
                ;       get memory address for current line
                ;
F5DF =          vladd   equ     $
F5DF DB98               in      vrlin           ;current board line
                vladd2:
F5E1 2100E8             lxi     h,vbase         ;start of board memory
F5E4 114001             lxi     d,140h          ;space between 4 lines
                vladd3:
F5E7 FE04               cpi     4               ;whittle down to < 4
F5E9 DAF2F5             jc      vladd4
F5EC 19                 dad     d               ;next 4 lines
F5ED D604               sui     4
F5EF C3E7F5             jmp     vladd3
                vladd4:
```

```
F5F2 111000             lxi     d,10h           ;space between single lines
F5F5 B7                 ora     a               ;down to 1
F5F6 C8                 rz
                vladd5:
F5F7 19                 dad     d               ;bump to next line
F5F8 3D                 dcr     a
F5F9 C2F7F5             jnz     vladd5
F5FC C9                 ret
                ;
                ;       add current char offset to ptr in hl
                ;
F5FD =          vcradd  equ     $
F5FD 114000             lxi     d,40h           ;space between groups
                vcradd1:
F600 FE10               cpi     16              ;whittle down to < 16
F602 DA0BF6             jc      vcradd2
F605 19                 dad     d               ;next group
F606 D610               sui     16
F608 C300F6             jmp     vcradd1
                vcradd2:
F60B 1600               mvi     d,0
F60D 5F                 mov     e,a
F60E 19                 dad     d               ;add to pointer
F60F C9                 ret
                ;
                ;       return current char address
                ;
F610 =          vcadd   equ     $
F610 CDDFF5             call    vladd           ;start of line address
F613 DB99               in      vrchr           ;board cursor
F615 3D                 dcr     a               ;relative 0
F616 CDFDF5             call    vcradd          ;add char offset
F619 C9                 ret
                ;
                ;       video driver special char table
                ;
F61A =          vtbl    equ     $
F61A 0B                 db      0bh
F61B 89F4               dw      vclear
F61D 17                 db      17h
F61E A6F4               dw      vup
F620 1A                 db      1ah
F621 BFF4               dw      vdown
F623 01                 db      01h
F624 D8F4               dw      vleft
F626 13                 db      13h
F627 E5F4               dw      vright
F629 0E                 db      0eh
F62A 98F4               dw      vhome
F62C 0D                 db      0dh
F62D F2F4               dw      vcr
F62F 0A                 db      0ah
F630 F7F4               dw      vlf
F632 7F                 db      7fh
```

```
F633 32F5               dw      vbacks
F635 1B                 db      1bh
F636 54F5               dw      vesc
F63E 05                 db      5
F639 3BF5               dw      veeol
F63B 00                 db      0
F63C 0000               dw      0
                ;
                ;       scratch area
                ;
EF80 =          vscrtch equ     vbase+vmxlin*vmxchr
EF80 =          vclrscn equ     vscrtch
EF81 =          vspeedl equ     vscrtch+1
EF82 =          vspeed  equ     vscrtch+2
EF83 =          vescflg equ     vscrtch+3
EF84 =          vphyl0  equ     vscrtch+4
EF85 =          vlfclr  equ     vscrtch+5
F63E                    end
```